PA-RISC 2.0 Architecture

This document is an excerpt from PA-RISC 2.0 ARCHITECTURE, by Gerry Kane, published by Prentice Hall PTR, isbn 0-13-182734-0. Print copies of the book can be ordered at www.hp.com/hpbooks.

This document contains:
   Chapter 11: Performance Monitor Coprocessor
11 Performance Monitor Coprocessor

The performance monitor coprocessor is an optional, implementation-dependent coprocessor which provides a minimal common software interface to implementation-dependent performance monitor hardware.

The performance monitor coprocessor responds to coprocessor instructions with a *uid* equal to 2.

**Performance Monitor Instructions**

The performance monitor instruction set consists of two instructions, PERFORMANCE MONITOR ENABLE (PMENB) and PERFORMANCE MONITOR DISABLE (PMDIS), which provide a common software interface to enable and disable the implementation-dependent performance monitor features.

The following figure shows the format of these operations and Table 11-1 shows the operations, their mnemonics, and sub-opcodes:

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Sub-op</th>
<th>Mnemonic</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0C</td>
<td>1</td>
<td>PMDIS</td>
<td>Disable performance monitor</td>
</tr>
<tr>
<td>0C</td>
<td>3</td>
<td>PMENB</td>
<td>Enable performance monitor</td>
</tr>
<tr>
<td>0C</td>
<td>0,2,4..F</td>
<td>undefined</td>
<td></td>
</tr>
<tr>
<td>0C</td>
<td>10..1F</td>
<td>reserved</td>
<td></td>
</tr>
</tbody>
</table>

The performance monitor coprocessor instructions are described at the end of this chapter.

When a performance monitor coprocessor instruction is executed and CCR[2] is 0, the coprocessor instruction causes an assist emulation trap. It is an undefined operation to set CCR[2] to 1 if the performance monitor coprocessor is nonexistent.

**Performance Monitor Interruptions**

Interruption vector number 29 in interruption group 2 is defined as the performance monitor coprocessor interrupt for implementation-dependent use by the performance monitor coprocessor. The interrupt is unmasked when the PSW F-bit is 1, and is masked when the PSW F-bit is 0. See Chapter 5, “Interruptions” for additional details.

**Reserved Sub-Opcode Exception**

When a performance monitor coprocessor instruction has a reserved sub-opcode, the implementation must signal a reserved-op exception by taking an assist exception trap.
Monitor Units

The monitor units are hardware units used to collect the necessary information during performance monitoring. The number of the monitor units and their hardware types are implementation dependent.

If a monitor unit provides counters, the most significant bit of the counter is required to be an overflow indicator. The bit must be set when the counter overflows and must remain set until explicitly reset by software. When the overflow indicator is set the remaining bits of the counter are undefined.

**NOTE**

If counters are used to implement the measurement units, it is recommended that the counters be at least 32 bits wide.
Performance Monitor Disable

Format: PMDIS,n

(55) | 0C | rv | 1 | 2 | n | rv |
---|----|----|---|---|---|----|
 6 | 12 | 5  | 3 | 1 | 5

Purpose: To disable the implementation-dependent performance monitor coprocessor, and conditionally nullify the following instruction.

Description: Disable all measurement units, after the current instruction. The following instruction is nullified if measurement is enabled and the .N completer is specified. The completer is encoded in the n field of the instruction.

Operation: if (n && measurement_enabled)
            PSW[N] ← 1;
            measurement_enabled ← 0;

Exceptions: Assist emulation trap
Performance Monitor Enable

Format: PMENB

<table>
<thead>
<tr>
<th>(55)</th>
<th>0C</th>
<th>rv</th>
<th>3</th>
<th>2</th>
<th>0</th>
<th>rv</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>12</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

Purpose: To enable the implementation-dependent performance monitor coprocessor.

Description: Enable the measurement units, starting with the next instruction.

Operation: measurement_enabled ← 1;

Exceptions: Assist emulation trap